2

In the Claims:

1.   (Previously Presented) A server adapted to communicate with a remote client, said server comprising:

a scene;

a plurality of object oriented programming (OOP) encapsulated virtual objects for use in an object-oriented environment; each said virtual object integrally having

at least a user-sensible aspect and further comprising at least a functional aspect; said user-sensible aspect being encapsulated as a user-sensible encapsulation, separately from said functional aspect, each said encapsulated virtual object being split by locating respective integrally related user-sensible and functional encapsulations of the same virtual object at said remote client and said server respectively, each said encapsulated virtual object comprising a definition of at least one relationship with another said encapsulated virtual object and said scene.

2.   (Previously Presented) A server as claimed in claim 1, said object-oriented environment being supported on a computer network comprising a first computer linked to a second computer; wherein the said user-sensible aspect is supported by said first computer and the said functional aspect is supported by said second computer.

3.   (Previously Presented) A server as claimed in claim 1, where said functional aspect is a behavioral aspect.

4.   (Previously Presented) A server as claimed in claim 1, where said user-sensible aspect comprises at least one of a display aspect and an interaction aspect.

5.   (Previously Presented) A server as claimed in claim 1, wherein said functional aspect is encapsulated in a functional encapsulation, and said functional encapsulation is exchangeable for an alternative functional encapsulation, thereby to alter the functionality of the said object.

3

6.   (Cancelled)

7.   (Previously Presented) A server as claimed in claim 1, wherein said at least one relationship is any one of a group comprising a coloring relationship, a positioning relationship, a shape relationship, a timing relationship, a movement relationship, a size relationship, a color relationship, a texture relationship and a reaction relationship.

8.   (Currently Amended) A~server~adapted~to~communicate~with~a~remote client, said server implementing a virtual computing environment, said virtual computing environment comprising:The server of claim 1, wherein at least some of said virtual objects comprise

a plurality of virtual objects, and a functionality for avoiding undesirable loops by preventing a consequential interaction between a first one of said virtual objects with a second one of said virtual objects when a number of interacting objects involved in said consequential interaction reaches a predefined maximum, wherein said interacting objects include said second virtual object, and any virtual object intermediating between said first virtual object and second virtual object.

9.   (Previously Presented) A server as claimed in claim 8, wherein said consequential interaction is direct.

10.   (Previously Presented) A server as claimed in claim 8, wherein said consequential interaction is an indirect interaction, being an interaction involving at least one mediating interaction with at least one intermediate object.

11.   (Previously Presented) A server as claimed in claim 10, said relationship with at least said second virtual object being defined by an order number, said order number being equal to the number of consequentially interacting objects.

4

12.   (Previously Presented) A server as claimed in claim 11, having a predetermined interaction limit, and an interaction stopper operable to prevent further consequential interactions occurring once a number of interactions corresponding to said interaction limit has been reached.

13.   (Previously Presented) A server as claimed in claim 12, wherein said predetermined interaction limit is specific to at least one of an interaction order and an interaction type, and said interaction stopper is operable to stop interactions within said specificity.

14.   (Previously Presented) A server as claimed in claim 8, wherein said consequential interaction with said at least second object comprises a change in at least one of location, movement, shape, size, status, internal parameters, color and texture of said second object.

15.   (Previously Presented) A virtual reality environment comprising a scene and a plurality of object oriented programming (OOP) encapsulated virtual objects supported by a scene database, said scene database having at least a first OOP object having an interchangeable functional unit associated therewith, said first interchangeable functional unit comprising functionality for each said virtual object, said virtual reality environment configured to support a method for facilitating interaction by a plurality of users at a plurality of client terminals with each said encapsulated virtual object, each said encapsulated virtual object having display and interaction characteristics and functional characteristics, said method comprising:

encapsulating the display and interaction characteristics in a display part of each said virtual object encapsulating functional characteristics in a functional part of each said encapsulated virtual object, each said encapsulated virtual object comprises a definition of at least one relationship with another said virtual object and said scene;

downloading said display part of said first object to user client terminals, and

5

retaining said functional part of said first object at a remote location networked with said user client terminals, thereby facilitating splitting said first virtual object between two terminals.

16.   (Previously Presented) A virtual reality environment as claimed in claim 15, wherein said functionality for each said encapsulated virtual object comprises behavior.

17.   (Previously Presented) A virtual reality environment as claimed in claim 15, wherein said functionality for each said encapsulated virtual object comprises rules for determining allowable interactions therewith.

18.   (Original) A virtual reality environment as claimed in claim 15, wherein said functionality comprises rules for determining non-allowable interactions therewith.

19.   (Original) A virtual reality environment as claimed in claim 15, wherein said functionality comprises rules for restricting allowable interactions therewith.

20.   (Previously Presented) An encapsulated virtual reality environment as claimed in claim 15, wherein each said encapsulated virtual object comprises a user-sensible aspect and said user-sensible aspect is encapsulated separately from said interchangeable functional unit.

21.   (Previously Presented) A virtual reality environment as claimed in claim 20, wherein said user-sensible aspect comprises at least one of data for display of at least one of said plurality of encapsulated virtual objects and interaction features.

22.   (Previously Presented) A virtual reality environment as claimed in claim 15, wherein said interchangeable functional unit is interchangeable to alter the functionality of said virtual object.

6

23.   (Previously Presented) A virtual reality environment as claimed in claim 15, wherein said first interchangeable functional unit comprises object-specific functionality for a plurality of virtual objects.

24.   (Previously Presented) A virtual reality environment as claimed in claim 20, further comprising at least one second virtual object where said first virtual object comprises a relationship with at least one other virtual object.

25.   (Previously Presented) A virtual reality environment as claimed in claim 24, wherein said relationship is direct.

26.   (Previously Presented) A virtual reality environment as claimed in claim 24, said first virtual object having a relationship with said at least one second virtual object such that an interaction applied to said first virtual object causes a consequential interaction with said at least one second virtual object.

27.   (Previously Presented) A virtual reality environment as claimed in claim 24, wherein said relationship with said at least one second object is an indirect relationship, being a relationship involving at least one mediating interaction with at least one intermediate object.

28.   (Previously Presented) A virtual reality environment as claimed in claim 27, said relationship with said at least one second object being defined by an order number, said order number being equal to the number of consequentially interacting objects.

29.   (Previously Presented) A virtual reality environment as claimed in claim 28, having a predetermined interaction total, and an interaction limiter operable to stop further first order consequential interactions occurring when a number of first order interactions equaling said predetermined interaction total has been reached.

30.   (Previously Presented) A virtual reality environment as claimed in claim 26, having a predetermined interaction total, and an interaction limiter operable to stop further consequential interactions occurring when a number of interactions equaling said predetermined interaction total has been reached.

31.   (Previously Presented) A virtual reality environment as claimed in claim 26, wherein said consequential interaction with said at least one second object comprises a change in position of said second object.

32.   (Previously Presented) A virtual reality environment as claimed in claim 24, wherein said consequential interaction with said at least one second object comprises any one of a group comprising a coloring interaction, a positioning interaction, a shape interaction, a timing interaction, a movement interaction, a size interaction, a color interaction, a texture interaction, a status interaction a sale and an internal parameter changing interaction.

33.   (Previously Presented) A virtual reality environment as claimed in claim 24, wherein said relationship is dynamically defined by a logical query.

34.   (Currently Amended) The virtual reality environment of claim 15, further comprising A a dedicated control element for controlling the functionality of said virtual objects belonging to a set of virtual objects within a virtual reality environment, said dedicated control element being associated with said virtual reality environment, and comprising:

identification functionality for determining whether a first virtual object within said virtual reality environment is a member of said set, and

control functionality for processing events received from said first virtual object, said control functionality being operable to avoid undesirable loops by preventing a consequential interaction betweeen between said first one of said virtual objects and a second one of said virtual objects when the number of interacting objects involved in said consequential interaction reaches a predefined maximum, wherein said interacting objects

8

include said second virtual object, and any virtual object intermediating between said first virtual object and second virtual object.

35.   (Previously Presented) A method for facilitating interaction by a plurality of users at a plurality of client terminals with at least an object oriented programming (OOP) encapsulated first object, said first object having display and interaction characteristics and functional characteristics, said characteristics being integrally related within said encapsulated first object, in a networked virtual reality environment, said method comprising:

encapsulating the display characteristics in a display and interaction part of said first object,

encapsulating the functional characteristics in a functional part of said first object,

downloading said display and interaction part of said first object to user client terminals, and

retaining said functional part of said first object at a remote location networked with said user client terminals, thereby facilitating splitting said first virtual object between two terminals, said at least an encapsulated first object comprising a definition of at least one relationship with a second virtual object and said scene.

36-43. (Cancelled)

44.   (Previously Presented) A method for controlling the functionality of a set of object oriented programming (OOP) virtual objects within a virtual reality environment having a scene, comprising:

incorporating allowable functionality for said set of virtual objects within a dedicated control element associated with said virtual reality environment,

incorporating identification functionality within said dedicated control element to enable said dedicated control element to distinguish between virtual objects within said set and virtual objects not within said set, and

9

thereby allowing said dedicated control element to control virtual objects within said set, said control element comprising a method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first encapsulated virtual object, said virtual object comprising at least a user-sensible aspect and further comprising at least a functional aspect, said user-sensible aspect and said functional aspect being integrally related within said first encapsulated virtual object; said user-sensible aspect being encapsulated as a user-sensible encapsulation, separately from said functional aspect, thereby facilitating splitting said first virtual object between two terminals, each said virtual object comprises a definition of at least one relationship with at least one of said virtual object and said scene.

45.   (Previously Presented) A method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first object oriented programming (OOP) encapsulated virtual object, said first object having display characteristics and functional characteristics integrally related therein, in a networked virtual reality environment having a scene and a plurality of object oriented programming (OOP) virtual objects; said method comprising:

encapsulating the display characteristics in a display and interaction part of said first object;

encapsulating the respectively integrally related functional characteristics in a functional part of said first object;

downloading said display and interaction part of said first object to user client terminals, each said virtual object comprising a definition of at least one relationship with another said virtual object and said scene, and

retaining said functional part of said first object at a remote location networked with said user client terminals, thereby facilitating splitting said first virtual object between two terminals;

said interactions comprising trading using said objects.

10

46.   (Previously Presented) The server of claim 1, wherein said plurality of OOP encapsulated virtual objects are added and removed from said object-oriented environment in real time.